UNIVERSITY OF SOUTHERN CALIFORNIA

Observations on COTS Software Integration Effort Based on the COCOTS Calibration Database

Chris Abts USC-CSE

Dr. Barry Boehm USC-CSE

Dr. Betsy Clark SMI

25th Annual Software Engineering Workshop

NASA/Goddard Space Flight Center

29 November 2000

UNIVERSITY OF
SOUTHERN CALIFORNIA

USC Center for Software Engineering

Points of Contact at USC-CSE in Los Angeles

Mr. Chris Abts (primary graduate researcher)…………...………. ……(213) 740-6470
Ms. Ladonna Pierce (CSE Office Administrator)…..…………………..……(213) 740-5703
Dr. Barry W. Boehm (CSE Director)…………………………………..….(213) 740-8163
USC Center for Software Engineering FAX line……..…………….……(213) 740-4927
COCOTS E-Mail……………………………………………..……*cots-info@sunset.usc.edu*
World Wide Web page………..………..……*http://sunset.usc.edu/COCOTS/cocots.html*

Additional Contact at Software Metrics, Inc. in Virginia (near D.C.)

Dr. Elizabeth (Betsy) Clark……………………………..……..……..(703) 754-0115
FAX line…………………………………………………………….……(703) 754-0115
E-Mail……………………………………………..……*Betsy@Software-Metrics.com*

UNIVERSITY OF
SOUTHERN CALIFORNIA

Outline

- Introduction and Model Overview
- General Insights from COTS Calibration Data
- Latest Calibration Results
- Conclusion

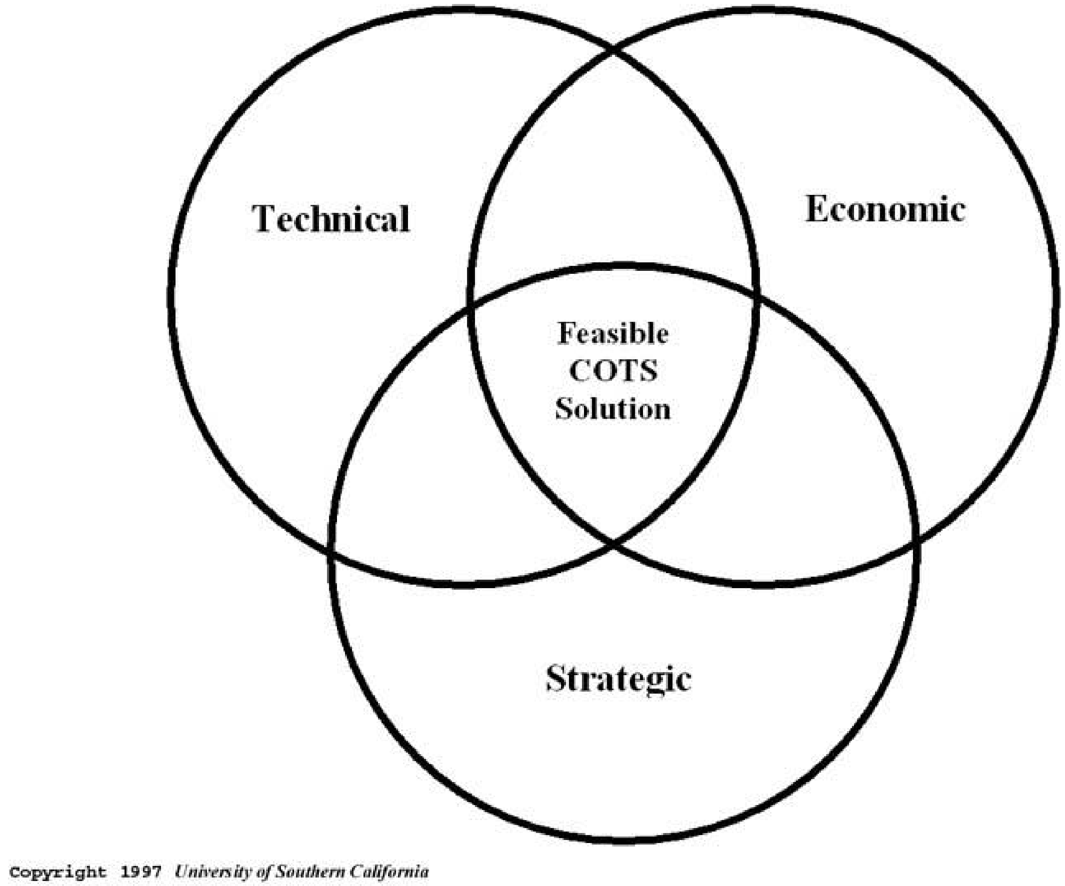
UNIVERSITY OF
SOUTHERN CALIFORNIA

Introduction and Model Overview

Consequences of Using COTS Products

- Two main characteristics of COTS:
  - source code not available to developer
  - evolution not under control of developer
- Results in trade-off:
  - development time can be reduced, but often at cost of increased s/w component integration/verification/validation work
- Unique risks associated with COTS:
  - cost of licensing and redistribution rights, royalties, effort needed to understand the COTS software, pre-integration assessment and evaluation, post-integration certification of compliance with mission critical or safety critical requirements, indemnification against faults or damage caused by vendor supplied components, and costs incurred due to incompatibilities with other needed software and/or hardware

When are COTS Products the “Right” Solution?

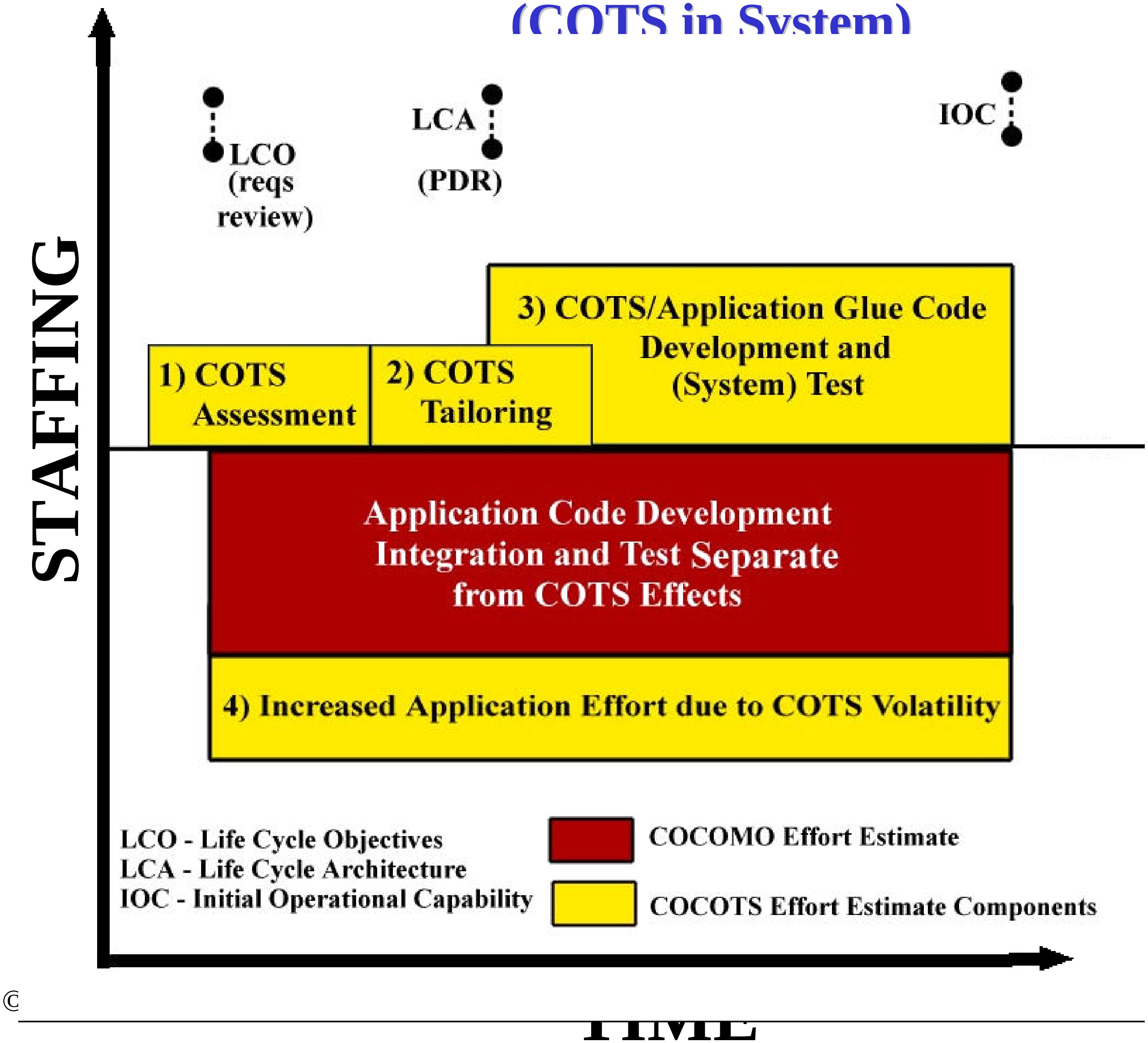


When are COTS Products the "Right" Solution?

- When they lie at the intersection of the three determinants of feasibility, *and do so demonstrably better than could original code*:
  - technical, economic, and strategic constraints
- Technical
  - ability to supply the desired functionality at the required level of reliability
- Economic
  - ability to be incorporated and maintained in the new system within the available budget and schedule
- Strategic
  - ability to meet needs of the system operating environment--including technical, political, and legal considerations--now, and as environment is expected to evolve in the future

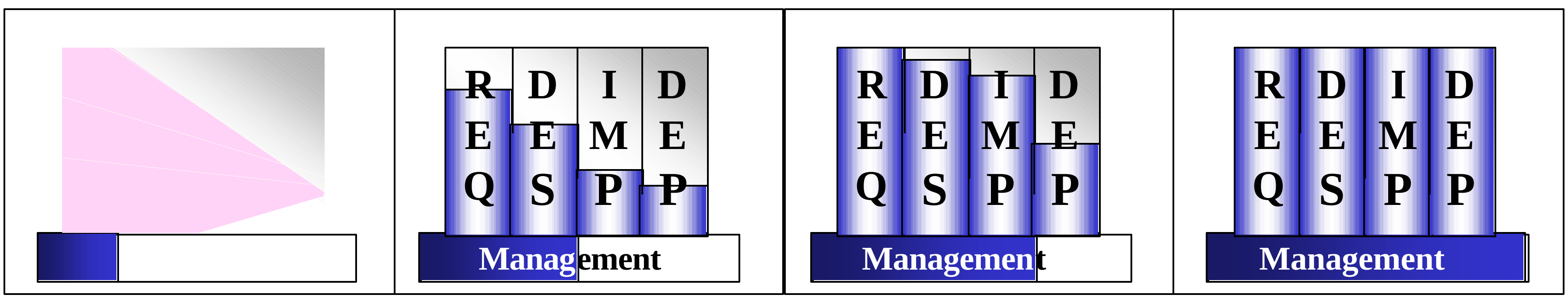
COCOMO vs. COCOTS Cost Sources

(COTS in System)

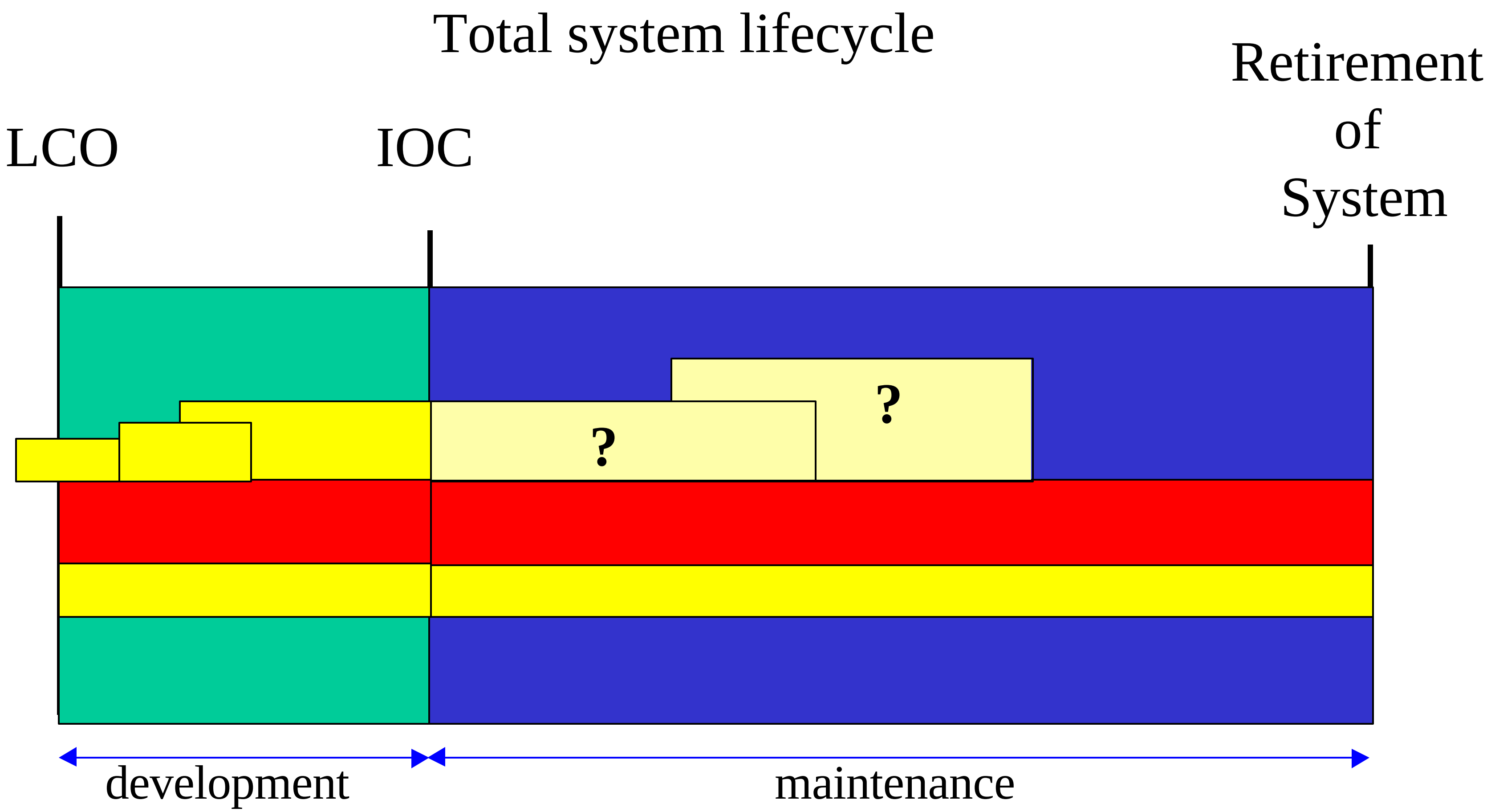


Objectory Information Set Evolution

Engineering Stage		Manufacturing Stage	
Inception	**Elaboration**	**Construction**	**Transition**
Feasibility Iterations	Architecture Iterations	Usable Iterations	Product Releases



COCOMO vs. COCOTS Cost Sources
(COTS in System)



UNIVERSITY OF
SOUTHERN CALIFORNIA

General Insights from COTS Calibration Data

Current COCOTS Database

- 20 Industrial projects
- Data collection continuing
  - (COCOMO 81 debuted with 63 calibration data points)
- Following summaries based on those 20 points

Project Domains

(project sources: Army, Navy, FAA, CSE Affiliates)

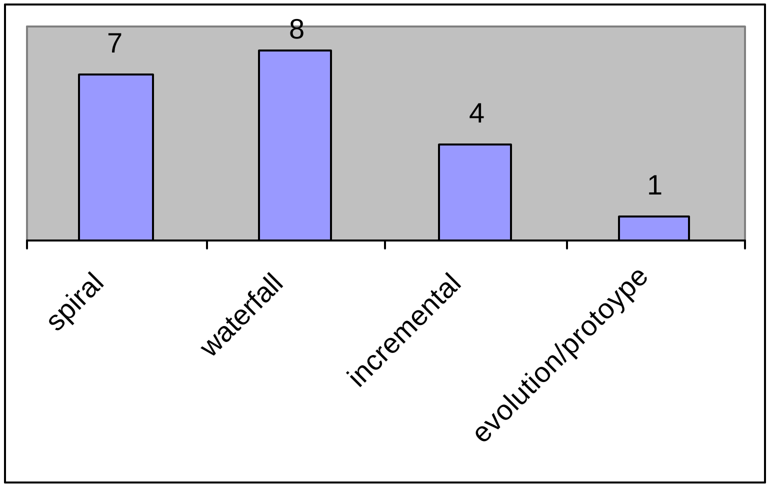
- Air Traffic Management 8
- Business (including databases) 3
- Communication, Navigation, & Surveillance 4
- Logistics 1
- Mission Planning 1
- Operations 2
- Web-based Maps 1

Classes of COTS Products Used

- configuration mgmt/build tools
- databases
- data conversion packages
- disk arrays
- compilers
- communication protocols/packages
- emulators
- engineering tools (reqs mgmt, design)
- software process tools
- GUIs/GUI builders
- graphic information systems
- middleware
- operating systems
- network managers
- device drivers
- problem mgmt
- report generators
- back office retail
- telecommunication & infrastructure
- telemetry analysis
- telemetry processing
- word processors

Development Processes

- Spiral 7
- Waterfall 8
- Incremental 4
- Evolution/Prototype 1



(most projects currently in maintenance)

UNIVERSITY OF SOUTHERN CALIFORNIA

Architectures & Architecting Processes

- Architectures
  - *all over the map but with two highly common elements:*
    "distributed" and "multi-threaded"
- Process *(some projects used more than one)*

  1- demos/prototypes 11

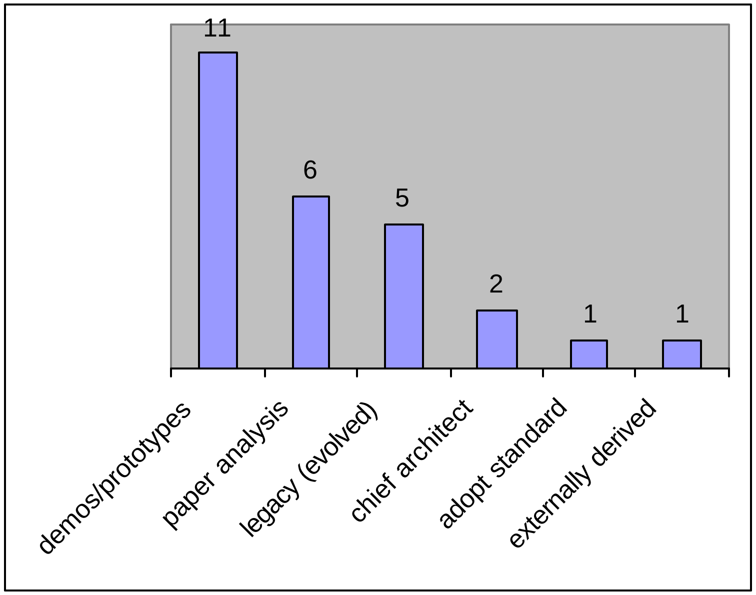
  2- paper analysis 6

  3- (evolved) legacy 5

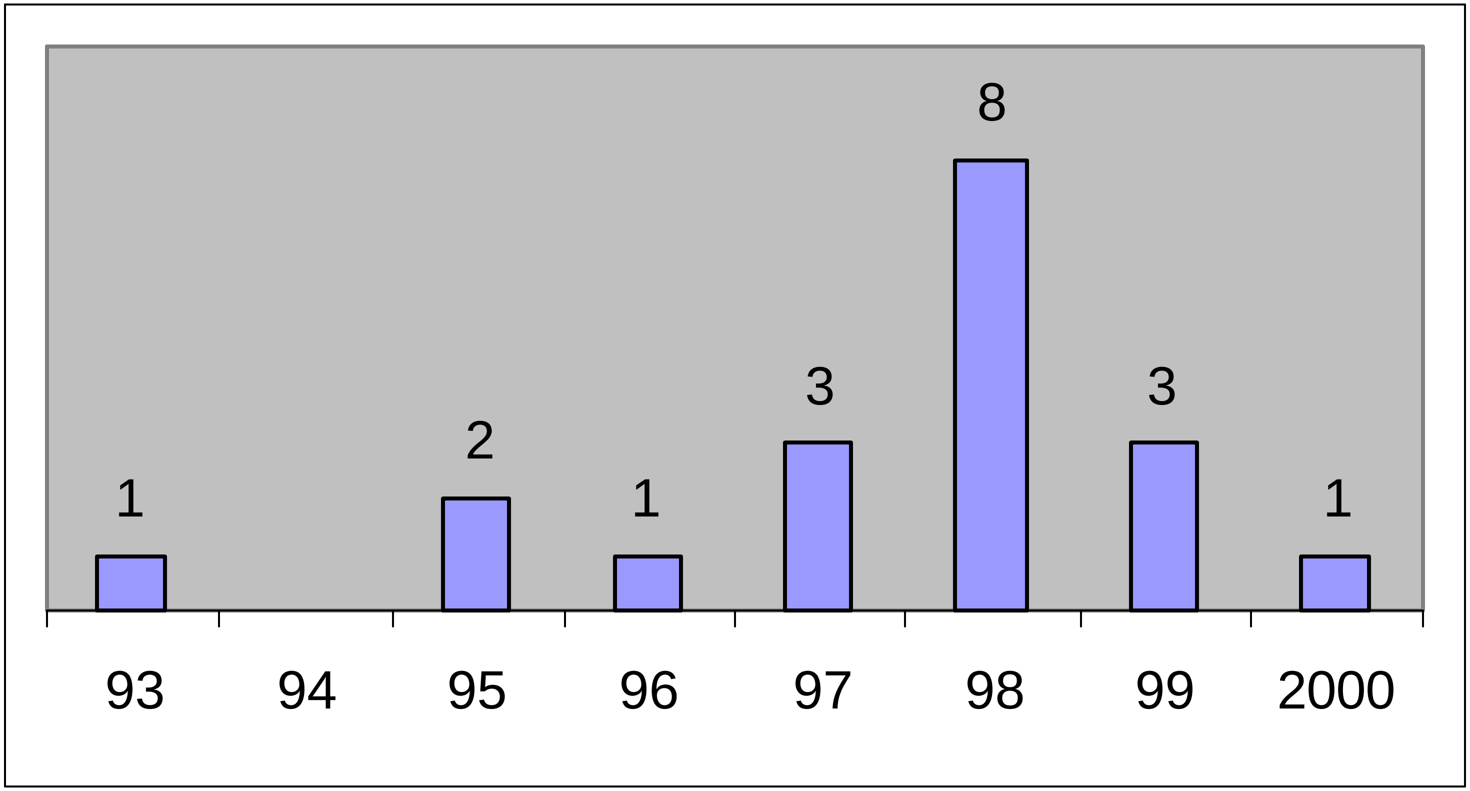
  4- chief architect 2

  5- adopt industry standard 1

  6- externally developed 1

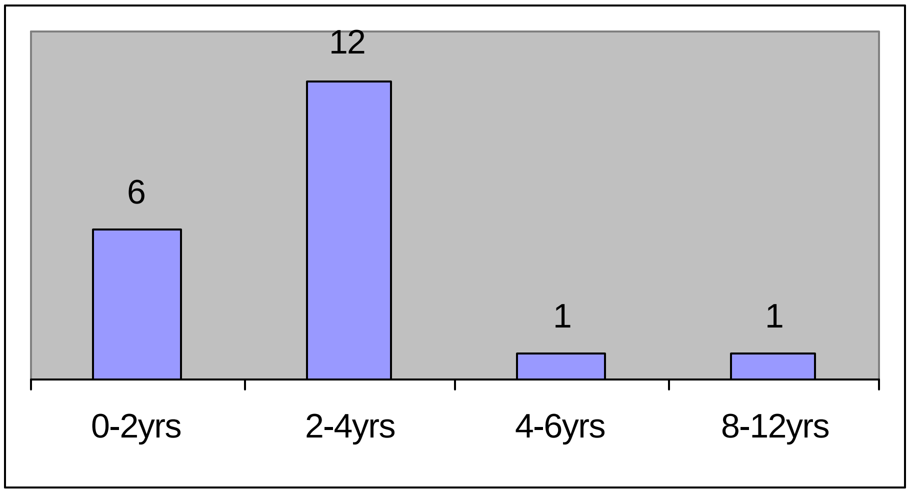


Delivery Dates

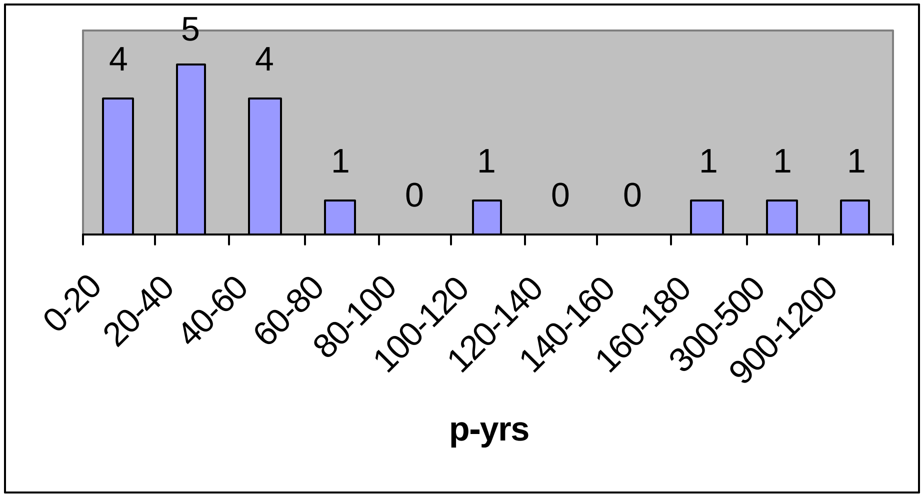


UNIVERSITY OF SOUTHERN CALIFORNIA

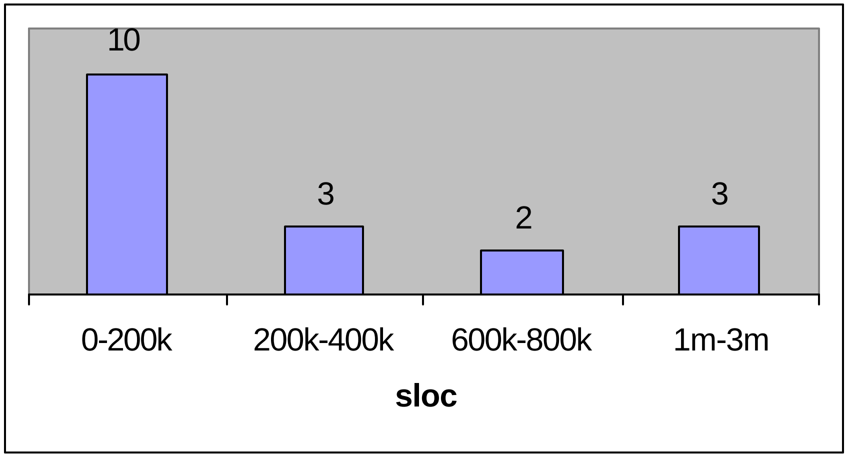
Total Duration



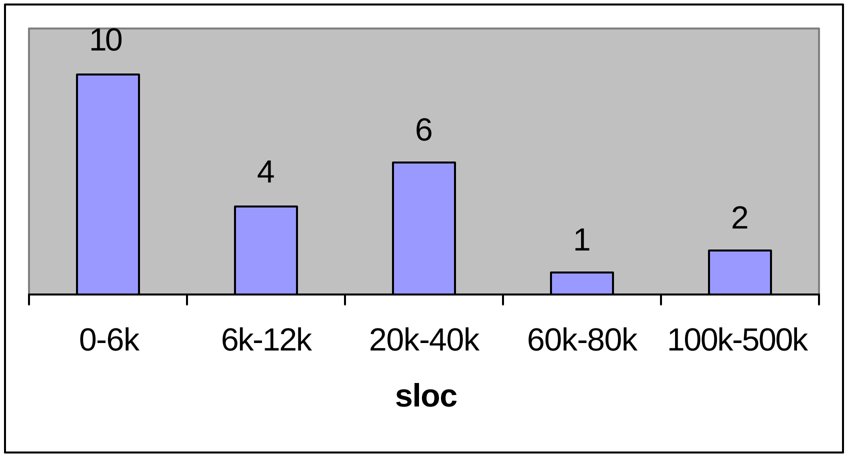
Total Effort



Total SLOC



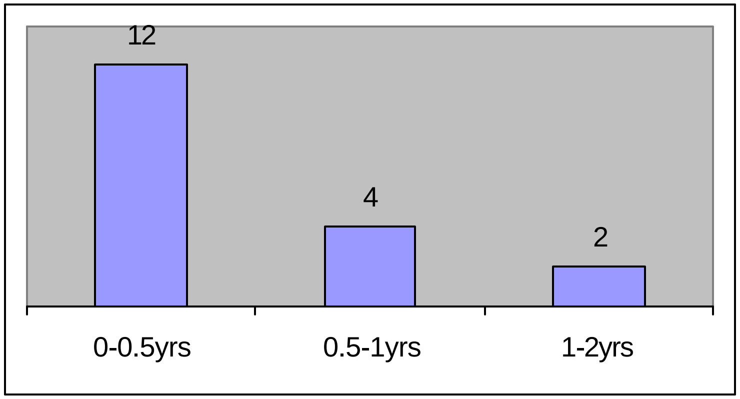
Glue SLOC



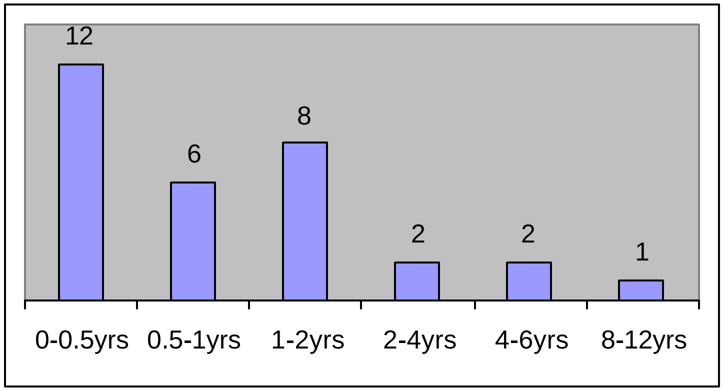
UNIVERSITY OF SOUTHERN CALIFORNIA

Schedule Duration by Activity

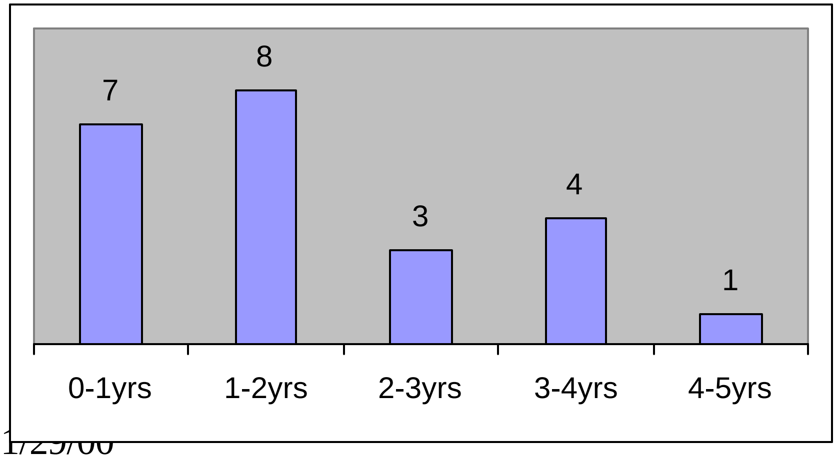
Assessment



Tailoring

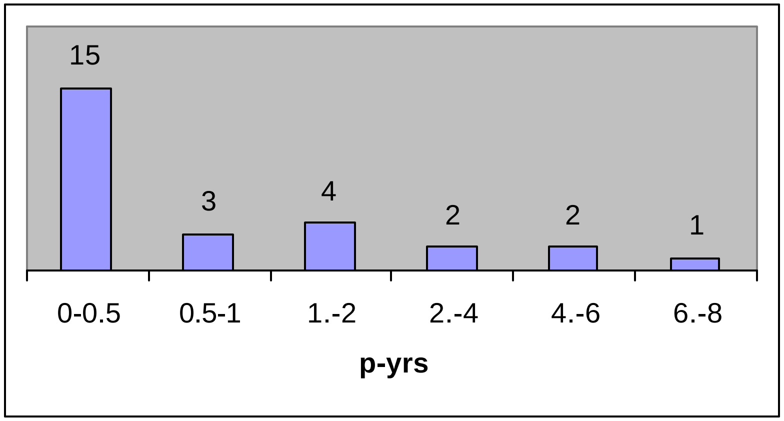


Glue Code

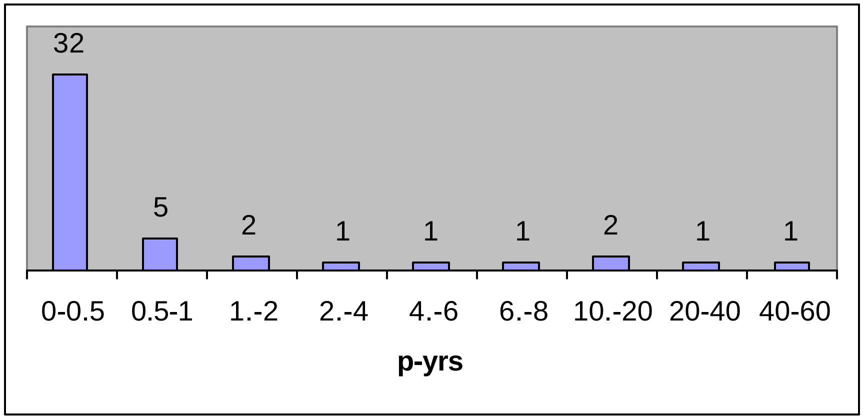


Effort by Activity

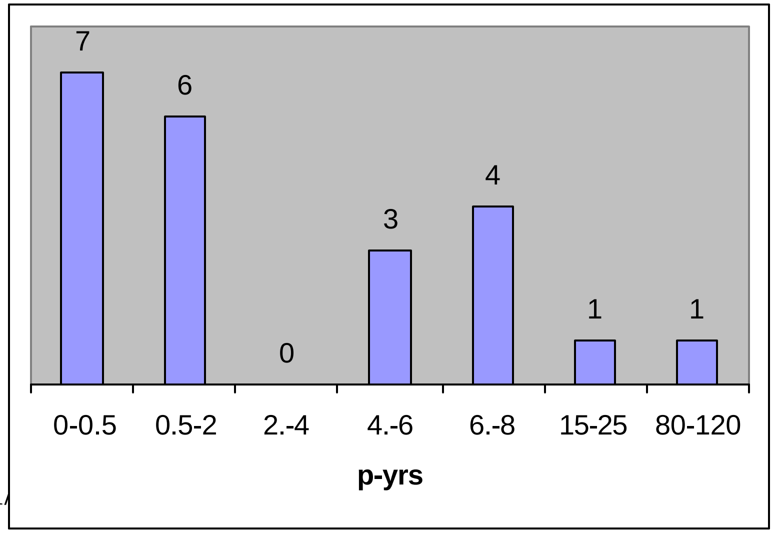
Assessment



Tailoring

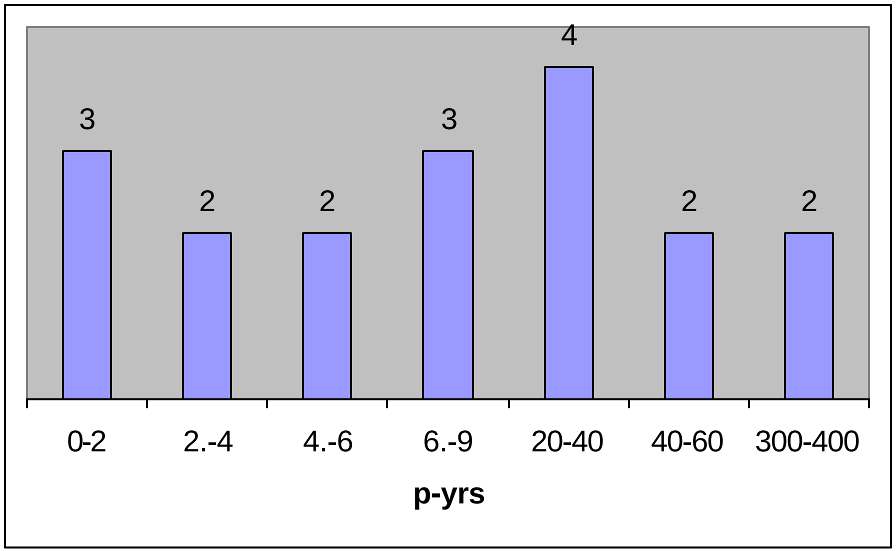


Glue Code

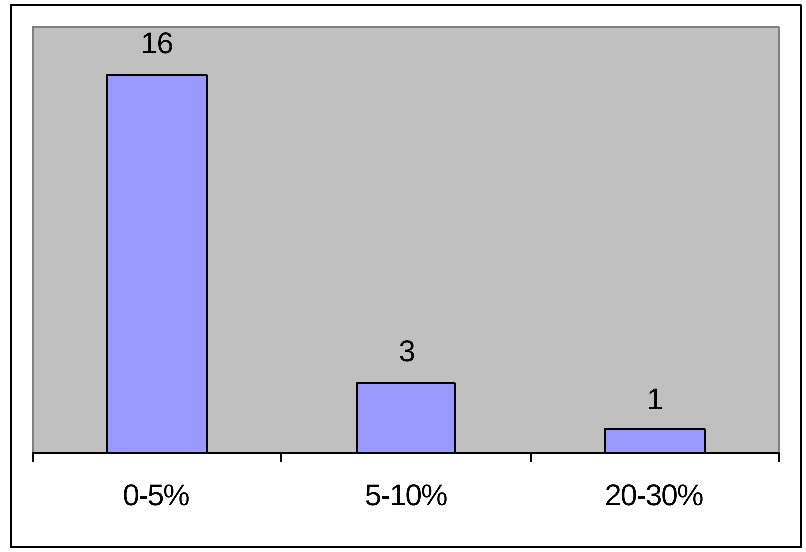


Effort by Activity *(cont'd)*

System effort due to COTS volatility

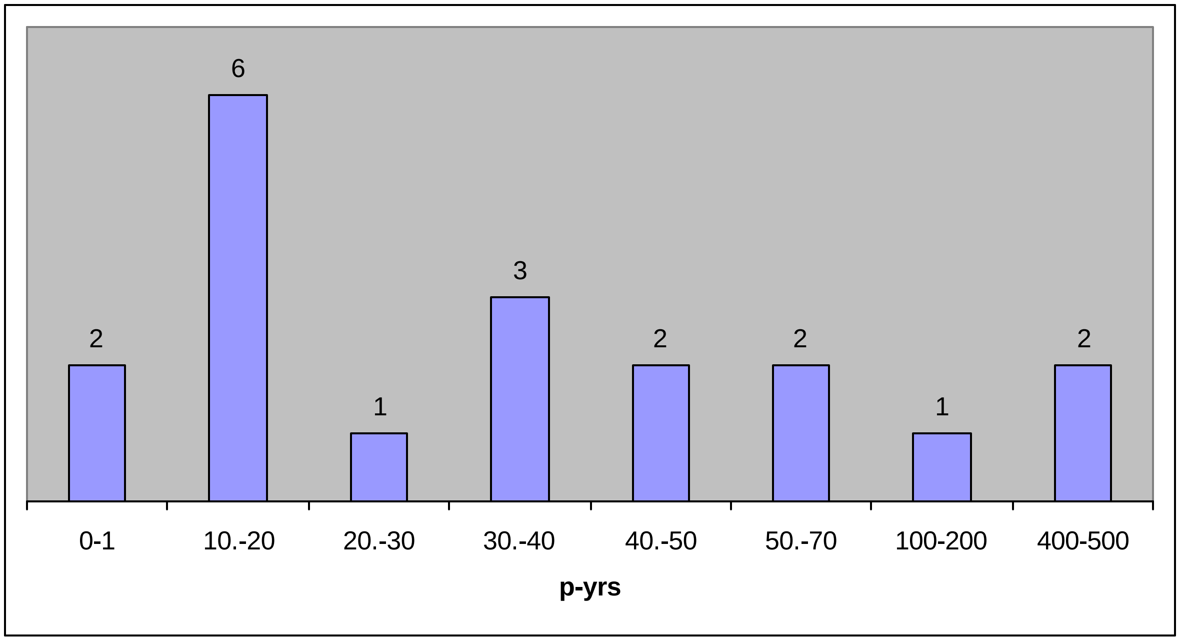


Percentage system rework due to COTS volatility



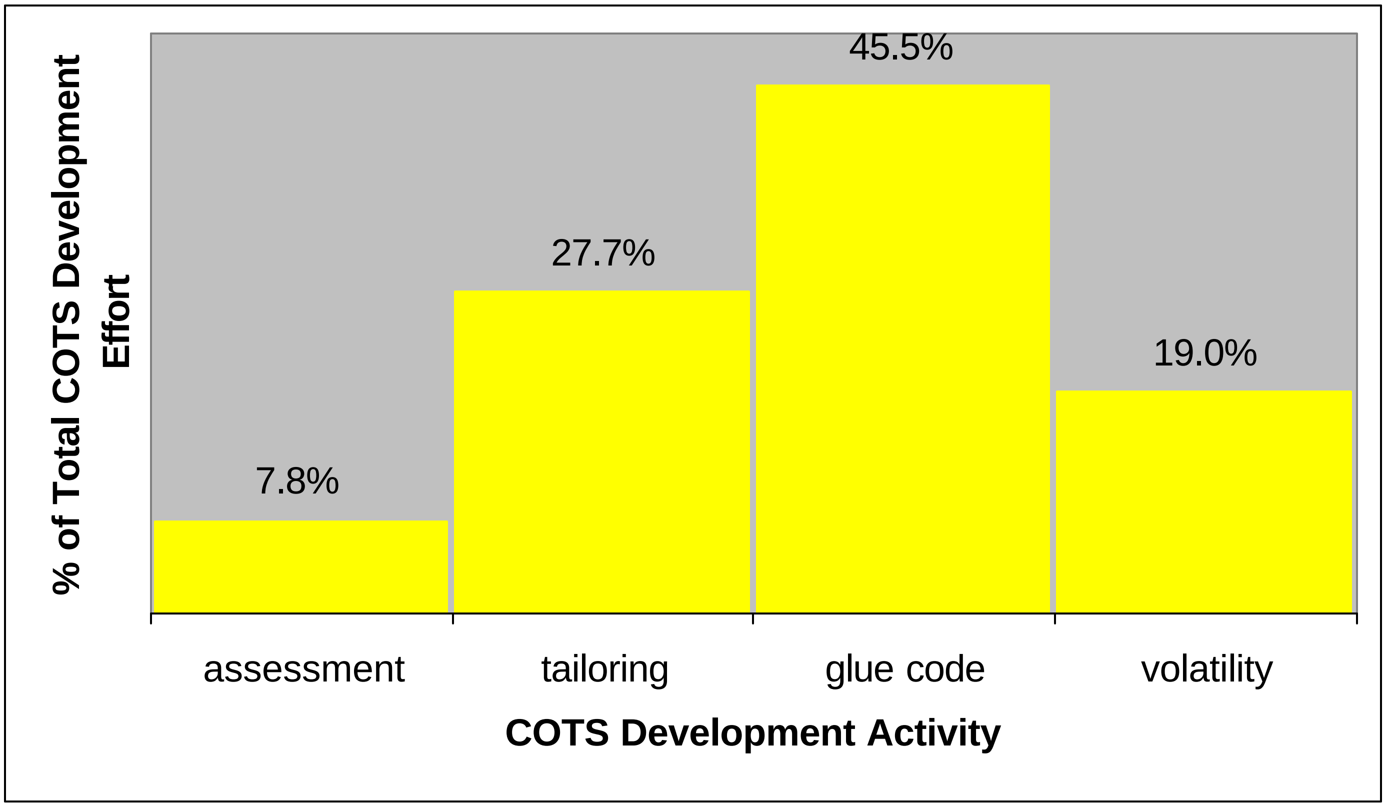
Total COTS Effort in System

(assessment+tailoring+glue code+volatility)



UNIVERSITY OF SOUTHERN CALIFORNIA

Allocation by percentage of gross reported effort across all project data points: the beginnings of an *empirically*-based, initial COTS development effort distributed by activity



Latest Calibration Results

UNIVERSITY OF SOUTHERN CALIFORNIA

10.23.2000 Total Program COTS Integration Effort

	Total FPAE (PM)				Total PTE (PM)				Total Glue Code (PM)		
Program	**Est.**	**Rept.**	**%Rerr**	**Program**	**Est.**	**Rept.**	**%Rerr**	**Program**	**Est.**	**Rept.**	**%Rerr**
A	110.998	87	27.58%	A	324.3163	268	21.01%	A	4.35	12	-63.75%
B	17.25	40	-56.88%	B	2.07	3	-31.00%	B	711.82	60	1086.36%
C	19.584	4	389.60%								
D	45.993	69	-33.34%	D	12.67	12	5.58%	D	4.94	74	-93.32%
								E	196.43	250	-21.43%
G	12	7	71.43%	G	8	11	-27.27%	G	72.08	84	-14.19%
H	5.001	6	-16.65%	H	21.96	26	-15.54%	H	0.79	6	-86.90%
I	17.227	21	-17.97%								
J	14.5	24	-39.58%	J	38.01	38	0.03%	J	0.83	1	-17.27%
K	24.329	18	35.16%	K	18.15	90	-79.83%	K	1.58	4	-60.49%
L	7.503	7.5	0.04%	L	50.82	186	-72.68%	L	1.78	7	-74.64%
M	1.248	1.25	-0.16%	M	3	3	0.00%	M	38.76	81	-52.15%
N	9.668	4.5	114.84%	N	28.15	16	75.94%	N	16.94	12	41.14%
				O	26	12	116.67%	O	595.03	1411	-57.83%
				P	137.79	42	228.07%	P	89.39	96	-6.89%
Q	13.386	6	123.10%	Q	4	4	0.00%	Q	78.50	72	9.03%
R	29.931	147.5	-79.71%	R	46.3309	640	-92.76%				
S	68.53	58	18.16%	S	61.3228	183	-66.49%	S	29.84	18	65.79%
				T	11.32	12	-5.67%				
				U	6	50	-88.00%	U	122.74	75	63.65%

UNIVERSITY OF SOUTHERN CALIFORNIA

10.23.2000 Total Program COTS Integration Effort

	Total COTS IE (PM)		
Program	**Est.**	**Rept.**	**%Rerr**
R	76.26	787.5	-90.32%
L	60.10	200.5	-70.03%
K	44.06	112	-60.66%
D	63.61	155	-58.96%
O	621.03	1423	-56.36%
M	43.01	85.25	-49.55%
S	159.69	259	-38.34%
H	27.75	38	-26.98%
E	196.43	250	-21.43%
I	17.23	21	-17.97%
J	53.34	63	-15.34%
G	92.08	102	-9.73%
T	11.32	12	-5.67%
U	128.74	125	2.99%
Q	95.89	82	16.93%
A	439.66	367	19.80%
P	227.18	138	64.62%
N	54.75	32.5	68.48%
C	19.58	4	389.60%
B	731.14	103	609.84%

10 out of 20 (50%) of COTS Integration estimates within (+/-) 40% of actuals.

In Conclusion: COCOTS' Most Important Contribution to Date

- COCOTS is completely open. Regardless of whatever estimates it provides, the descriptions of the elements that have gone into the model help highlight the most important factors that should be of concern to managers and developers of software systems using COTS software components.

- It's a true "constructive" cost model:
  - one that helps an estimator better *understand the complexities* of a given software job to be done
  - by being open permits the estimator to know exactly *why* a model gives the estimate it does